

## Lab-8

Doubly linked list.

```
struct node {  
    struct node *prev;  
    int data;  
    struct node *next;  
};
```

~~add node at head~~

```
void add-at-begin() {  
    struct node *ptr = NULL;  
    ptr = (struct node *) malloc(sizeof(struct node));  
    printf("enter the node data");  
    scanf("%d", &ptr->data);  
    ptr->prev = NULL;  
    ptr->next = NULL;  
    if (head == NULL) {  
        head = ptr;  
    }  
    else {  
        ptr->next = head;  
        head->prev = ptr;  
        head = ptr;  
    }  
}
```

```
void delete-at-specifiedloc() {  
    int i, loc;  
    if (head == NULL) {  
        printf("empty list");  
    }  
}
```

else {

    struct node \*ptr = head;

    printf("enter the location: ");

    scanf("%d", &loc);

    for(i=1; i<loc; i++){

        ptr = ptr->next;

    }

    ptr->prev->next = ptr->next;

    ptr->next->prev = ptr->prev;

    free(ptr);

}

}

void display() {

    if(head == NULL) {

        printf("list is empty\n");

    }

    else {

        struct node \*temp = head;

        while(temp != NULL) {

            printf("%d\t", temp->data);

            temp = temp->next;

        }

        printf("\n");

    }

}